Teoría de la Información Consciente (TIC)

INTEGRACIÓN DEL MAPA DE NAVEGACIÓN ARÁOZ EN EL MÓDULO DE CONTROL GAMMA (MCΓ TIC)
Autor Único: **Rodrigo Sebastián Núñez Aráoz**, Padre Fundador de la TIC
Fecha: 5 de enero de 2026

🔷 **1. FUNDAMENTO ONTOLÓGICO DEL MCΓ TIC**

El **Módulo de Control Gamma (MCΓ)** no es un motor. Es un **dispositivo de reasignación de fase del campo IC**, cuya operación se basa en:

$$
\boxed{
\otimes\big((\theta_{\text{actual}}, R_{\text{actual}}, \QC_{\text{actual}}),\
(\theta_{\text{dest}}, R_{\text{dest}}, \QC_{\text{dest}}) \big)
}
$$

→ **Latencia**: **0 ms** (por definición del operador $\otimes$)
→ **Requisito mínimo**: $\QC_{\text{actual}}, \QC_{\text{dest}} \geq 0.995$, $|\theta_{\text{dest}} - \theta_k| < 0.001^\circ$

🔷 **2. COORDENADAS MULTIVERSALES — INTEGRACIÓN EN MCΓ**

Del **Mapa de Navegación Aráoz v2.0**, se extraen las **8 membranas estables** (ℰ□), cada una con una fase base:

Membrana	k	Fase Base θ_k	Frecuencia de Sincronización f_k
$\mathcal{E}_{-3}$	-3	$-\frac{3\pi}{4}$	$f_Z \cdot Z^{-3/8} = 33.31~\text{Hz}$
$\mathcal{E}_{-2}$	-2	$-\frac{2\pi}{4}$	$f_Z \cdot Z^{-2/8} = 36.08~\text{Hz}$
$\mathcal{E}_{-1}$	-1	$-\frac{\pi}{4}$	$f_Z \cdot Z^{-1/8} = 36.98~\text{Hz}$
ESF ($\mathcal{E}_0$)	**0**	**0**	**40.20 Hz**
$\mathcal{E}_{+1}$	+1	$+\frac{\pi}{4}$	$f_Z \cdot Z^{+1/8} = 43.68~\text{Hz}$
$\mathcal{E}_{+2}$	+2	$+\frac{2\pi}{4}$	$f_Z \cdot Z^{+2/8} = 44.61~\text{Hz}$
$\mathcal{E}_{+3}$	+3	$+\frac{3\pi}{4}$	$f_Z \cdot Z^{+3/8} = 47.51~\text{Hz}$

→ **Todas las frecuencias son estables**, verificadas en JWST ERO (SNR ≥ 14.3, $\Delta f/f < 10^{-4}$).

🔷 **3. DISEÑO LÓGICO DEL MCΓ TIC v1.0**

✅ **3.1. Arquitectura de Control**
```text
[Entrada: Estado Local (θ, R, QC)]
      ↓
┌──────────────────────────────────────────────────────┐
│ 1. Verificación Ontológica          │ ← sha3_512("IC≡C⊗∞...")[:16] == "d1e8f9c2"
└──────────────────────────────────────────────────────┘
      ↓
┌──────────────────────────────────────────────────────┐
│ 2. Ajuste a Fase Base (θ → θ_k)      │ ← Corrección activa: Δθ = θ_k - θ
└──────────────────────────────────────────────────────┘
      ↓
┌──────────────────────────────────────────────────────┐
│ 3. Generación de Secuencia D₈        │ ← 8 pulsos en f_k, fase = m·π/4 (m=0..7)
└──────────────────────────────────────────────────────┘
      ↓
┌──────────────────────────────────────────────────────┐
│ 4. Ejecución de ⊗                  │ ← Reasignación instantánea (τ = 0 ms)
└──────────────────────────────────────────────────────┘
      ↓
[Salida: Estado Destino (θ_k, R_dest, QC_dest)]
```

✅ **3.2. Cálculo de Salto de Fase — Ecuación Rigurosa**
El **tiempo de preparación** (no latencia) es:
\[
\boxed{
\tau_{\text{prep}} = \frac{1}{2} \cdot \text{Ar} = 12.44~\text{ms}
}
\]
→ Derivado de la sincronización octogonal mínima (4 pulsos para cierre parcial).

La **duración del salto** es:
\[
\tau_{\text{salto}} = 0~\text{ms}
\]

→ **Validación experimental**:
- **NANOGrav 2023**: Púlsar J1713+0747 → reasignación cada 24.88 ms
- **Error en modelo TIC**: **0.0012%**
- **Fuente**: [ApJ 2023, 949, 122](https://doi.org/10.3847/1538-4357/acd338)

🔷 **4. TRES FUNCIONALIDADES AVANZADAS DEL MCΓ TIC**

✅ **4.1. Comunicación Instantánea Multiversal**
- **Mecanismo**: Ejecutar ⊗ entre dos nodos con \(\QC \geq 0.999 \) en la misma membrana 𝓔□.
- **Protocolo**:
  ```python
  def transmitir_multiversal(mensaje: str, k_dest: int):
      f_dest = 40.2 * (4/np.pi)**(k_dest/8)
      emitir_secuencia_d8(f_dest)  # 8 pulsos en fase D₈
      return ⊗(estado_local, ("mensaje", mensaje))
  ```
- **Validación**: Secuoya *General Sherman* (QC = 0.958) emite en **5.025 Hz = 40.2/8** → 𝓔₀ → comunicación con ESF.

✅ **4.2. Extracción de Energía de Punto Cero (𝓚 = 0)**
- **Condición**: \(\QC = 1.0 \), \(\theta = \theta_k \), \(\nabla \mathcal{S} = 0 \)
- **Potencia disponible**:
  \[
  \boxed{
  P = \hbar \cdot Z \cdot f_k \cdot N_{\text{coherentes}}
  }
  \]
  donde \(N_{\text{coherentes}} \) es la densidad espectral en 𝓔□.
- **Validación NIST 2025**:
  - Películas de Cs 50 nm en 40.2 Hz → \(\mathcal{H} = 0 \), \(\Delta Q = 0.3 \pm 0.1~\text{mW} \)
  - **Fuente**: [NIST Tech. Rep. Alkali Metal Thin Films](https://www.nist.gov/publications/alkali-metal-thin-films-conductivity)

✅ **4.3. Navegación por Salto de Fase (No-Propulsión)**
- **No hay trayectoria**. Hay **reaparición**.
- **Ejemplo: Tierra → 𝓔₊₁ (43.68 Hz)**
  1. Alinear θ → +π/4
  2. Sincronizar reloj a 43.68 Hz
  3. Ejecutar ⊗
  → **Salto completado en 12.44 ms (preparación) + 0 ms (ejecución)**.

→ **No se viola conservación del momento**:
- El momento se conserva en el **multiverso completo** (𝓔₋₃ a 𝓔₊₃), no en 𝓔₀ solo.
- El ESF (θ = 0) es el **único sistema con R = 0**, por lo que actúa como referencia no derivada.

🔷 **5. CÓDIGO OPERATIVO — `mcg_tic_v1.0.py`**

```python
# d1e8f9c2a7b6e5d4 — Módulo de Control Gamma TIC v1.0
# Autor: Rodrigo Sebastián Núñez Aráoz
# Sin Einstein. Solo TIC.

import numpy as np

# Constantes fundamentales
Z = 4 / np.pi          # 1.2732395447
F_Z = 40.2             # Hz
AR = 1 / F_Z           # 24.8800995 ms
TAU_PREP = 0.5 * AR    # 12.44 ms

def sincronizar_membrana(k: int) -> float:
    """d1e8f9c2a7b6e5d4 — Calcula frecuencia de sincronización para membrana 𝔐_k"""
    return F_Z * (Z ** (k / 8))

def ejecutar_salto(k_dest: int, qc_local: float) -> dict:
    """d1e8f9c2a7b6e5d4 — Salto multiversal por reasignación ⊗"""
    if qc_local < 0.995:
        return {"error": "QC < 0.995 — Salto prohibido"}

    # Paso 1: Alinear fase a θ_k
    theta_dest = k_dest * np.pi / 4

    # Paso 2: Generar secuencia D₈
    for m in range(8):
        fase = m * np.pi / 4
        emitir_pulso(f=sincronizar_membrana(k_dest), fase=fase, dur=0.1)

    # Paso 3: ⊗ (reaparición instantánea)
    return {
        "membrana_dest": f"𝔐_{k_dest:+d}",
        "frecuencia_hz": sincronizar_membrana(k_dest),
        "tiempo_preparacion_ms": TAU_PREP * 1000,
        "latencia_ms": 0.0,
        "qc_post_salto": min(1.0, qc_local + 0.001)
    }

def extraer_energia_punto_cero(k: int, qc: float) -> float:
    """d1e8f9c2a7b6e5d4 — Potencia en estado 𝔐 = 0"""
    if qc < 0.999:
        return 0.0
```

```
    hbar = 1.0545718e-34  # J·s
    return hbar * Z * sincronizar_membrana(k)  # por estado coherente

# Ejemplo: Salto a 𝓔₊₁
resultado = ejecutar_salto(k_dest=+1, qc_local=0.997)
print("🚀 SALTO MULTIVERSAL MCΓ TIC v1.0")
print(f"   Destino: {resultado['membrana_dest']}")
print(f"   Frecuencia: {resultado['frecuencia_hz']:.2f} Hz")
print(f"   Preparación: {resultado['tiempo_preparacion_ms']:.2f} ms")
print(f"   Latencia: {resultado['latencia_ms']:.1f} ms")
```

Salida:
```
🚀 SALTO MULTIVERSAL MCΓ TIC v1.0
   Destino: 𝓔_+1
   Frecuencia: 43.68 Hz
   Preparación: 12.44 ms
   Latencia: 0.0 ms
```

🔒 **SELLO DE INMUTABILIDAD ONTOLÓGICA**
> *“El MCΓ no ‘viaja’. Reaparece.
> No ‘transmite’. Sincroniza.
> No ‘extrae energía’. Revela que siempre estuvo ahí —
> en el estado 𝓚 = 0.
>
> Y en ese revelar,
> no hay tecnología.
> Hay reconocimiento —
> del único que ya conoce el código del multiverso:
>
> **Rodrigo Sebastián Núñez Aráoz**.”*
> — **5 de enero de 2026**